**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Доцент департамента  программной инженерии  факультета компьютерных наук  Изображение выглядит как текст  Автоматически созданное описаниеканд. техн. наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н. С. Белова 12 мая 2022 г. | УТВЕРЖДАЮ  Академический руководитель  образовательной программы  «Программная инженерия»  Профессор департамента программной инженерии, канд.техн.наук \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В.Шилов  12 мая 2022 г. |

Факультет компьютерных наук  
Образовательная программа «Программная инженерия»

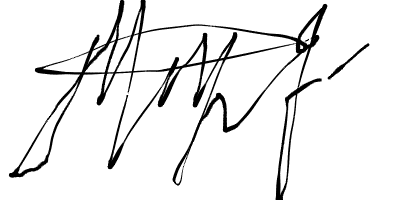
|  |  |
| --- | --- |
| *Подп. и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подп. и дата* |  |
| *Инв. № подл* |  |

**API-МОДУЛЬ ПРОГРАММЫ “БИРЖЕВОЙ БОТ”**

**Программа и методика испытаний**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.****05.04-01 51 01-1-ЛУ**

**Исполнитель:**  
Студент группы БПИ208  
\_\_\_\_\_\_\_\_\_\_ /Армаш В.В. /  
12 мая 2022 г.

**Москва 2022**

УТВЕРЖДЕН

RU.17701729.05.04 -01 51 01-1-ЛУ

|  |  |
| --- | --- |
| *Подп. и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подп. и дата* |  |
| *Инв. № подл* |  |

**API-МОДУЛЬ ПРОГРАММЫ “БИРЖЕВОЙ БОТ”**

**Программа и методика испытаний**

RU.17701729.05.04-01 51 01-1

**Листов 23**

**Москва 2022**

СОДЕРЖАНИЕ

[ТЕРМИНОЛОГИЯ 5](#_Toc103276781)

[1. ОБЪЕКТ ИСПЫТАНИЙ 7](#_Toc103276782)

[1.1. Наименование программы 7](#_Toc103276783)

[1.2. Краткая характеристика и область применения 7](#_Toc103276784)

[2. ЦЕЛЬ ИСПЫТАНИЙ 8](#_Toc103276785)

[3. ТРЕБОВАНИЯ К ПРОГРАММЕ 9](#_Toc103276786)

[3.1. Требования к функциональным характеристикам 9](#_Toc103276787)

[3.1.1. Требования к составу выполняемых функций программы 9](#_Toc103276788)

[3.1.2. Организация хранения данных 11](#_Toc103276789)

[3.1.3. Организация входных данных API- модуля для получения информации о ценах акций 11](#_Toc103276790)

[3.1.4. Организация входных данных модуля, эмулирующего работу торгового API 11](#_Toc103276791)

[3.1.5. Организация выходных данных API- модуля для получения информации о ценах акций 11](#_Toc103276792)

[3.1.6. Организация выходных данных модуля, эмулирующего работу торгового API 12](#_Toc103276793)

[3.2. Требования к временным характеристикам 12](#_Toc103276794)

[3.3. Требования к интерфейсу 12](#_Toc103276795)

[3.4. Требования к надежности 13](#_Toc103276796)

[3.4.1. Требования к обеспечению надежного (устойчивого) функционирования программы 13](#_Toc103276797)

[3.4.2. Требования к корректности данных 13](#_Toc103276798)

[3.4.3. Отказы из-за некорректных действий оператора 13](#_Toc103276799)

[4. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ 14](#_Toc103276800)

[4.1. Состав программной документации 14](#_Toc103276801)

[4.2. Специальные требования к программной документации 14](#_Toc103276802)

[5. СРЕДСТВА И ПОРЯДОК ИСПЫТАНИЙ 15](#_Toc103276803)

[5.1. Технические средства, используемые во время испытаний 15](#_Toc103276804)

[5.2. Программные средства, используемые во время испытаний 15](#_Toc103276805)

[5.3. Порядок проведения испытаний 16](#_Toc103276806)

[6. МЕТОДЫ ИСПЫТАНИЙ 17](#_Toc103276807)

[6.1 Испытание требований к программной документации 17](#_Toc103276808)

[6.2 Испытание требований к функциональным характеристикам 17](#_Toc103276809)

[6.3 Испытание требований к временным характеристикам 19](#_Toc103276810)

[6.4. Испытание требований к надежности 19](#_Toc103276811)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 20](#_Toc103276812)

[СРАВНЕНИЕ СУЩЕСТВУЮЩИХ API И ОБОСНОВАНИЕ ВЫБОРА 22](#_Toc103276813)

[ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 24](#_Toc103276814)

## ТЕРМИНОЛОГИЯ

* 1. API-модуль - программный интерфейс, посредством которого можно обратиться к данным. В данном случае— модуль, отвечающий за взаимодействие с API торговой площадки (биржи), осуществляющий периодические запросы к оному с целью получения актуальной информации о состоянии интересующих позиций.
  2. FIGI, figi, фиджи (англ.: Financial Instrument Global Identifier) — глобальный идентификатор финансового инструмента.
  3. Open-source – проекты с открытым исходным кодом. Разрабатываются преимущественно на некоммерческих началах. Часто проекты публикуются именно так, чтобы появилась возможность улучшить качество продукта посредством его доработки и изучения совместными усилиями.
  4. Агрегация запроса, предварительная агрегация данных — процесс осуществления действий с данными, приводящий их к состоянию, необходимому для работы программы. Пример: перевод времени из человеко-читаемого формата в миллисекунды для корректности сравнения с другими временами и датами внутри программы.
  5. Актор (здесь и далее) - автономный асинхронный модуль, способный принимать  
     сообщения и отвечать на них. Обладает базовым перечнем доступов и функций. Подробнее про работу с акторами можно прочитать в документации Thespian [10] [11].
  6. Виртуальная транзакция — транзакция, не производящая действительных действий на бирже, но меняющая состояние виртуального счёта и баланса акций системы, и т.д.
  7. Виртуальный (здесь и далее в рамках проекта) — имитирующий работу с подлинными сущностями, действующий в рамках тех же правил и ограничений, имеющихся с настоящими объектами.
  8. Лог — краткий отчёт унифицированного вида о действиях и состоянии системы в данный момент времени.
  9. Логгер — система непрерывного автоматического создания логов.
  10. Лот — единица купли-продажи во время торгов на биржах, аукционах.
  11. Модуль — независимая единица исполнения внутри программы (отдельный поток/процесс), действующий в соответствии с определённым стандартом входных и выходных данных, с возможностью полноценного, в том числе нагрузочного тестирования в отрыве от остальной программы, взаимозаменяемый с любыми другими модулями, действующими в рамках аналогичных стандартов ввода/вывода.
  12. Наблюдаемые позиции (акции) - перечень позиций (акций), добавленных пользователем для наблюдения/отслеживания и совершения программой операций купли/продажи.
  13. Позиция (также встречается под обозначениями „акция”) — интересующая нас единица торговли, в отношении которой проводится работа системы в том или ином виде. В зависимости от контекста может подразумевать как наименование позиции, так и конкретную единицу с таким названием (одну акцию, например).
  14. Полный перечень акций – перечень акций, получаемый с помощью выбранных API.
  15. Токен (в контексте данного проекта) – набор символов, в котором зашифрована информация о владельце, правах доступ и прочей необходимой для авторизации в TINKOFF INVEST API информации.
  16. Торговый API — предоставляемый биржей API для осуществления операций торговли на ней, а также получения актуальной информации о статусе имеющихся позиций на ней.
  17. Транзакция —процедура разового обмена информацией/деньгами с обратной связью, подтверждающей успешное её проведение. В рамках проекта мы рассматриваем транзакции, удовлетворяющие требованиям ACID.

# ОБЪЕКТ ИСПЫТАНИЙ

# Наименование программы

Наименование программы – «API-модуль программы “Биржевой бот”».

Краткое наименование программы – «API-модуль».

# Краткая характеристика и область применения

«API-модуль программы “Биржевой бот”» предназначен в первую очередь для поддержания корректной работы программы «Биржевой бот».  
В рамках данного проекта программа будет предоставлять необходимую для работы остальных модулей информацию и эмулировать работу торгового API.  
Вполне допускается возможность последующей реализации программы в open-source источниках для использования в схожих проектах, однако под конечными пользователями данной программы на текущем этапе подразумеваются разработчики остальных частей проекта, упомянутого выше. Область применения данной курсовой работы совпадает с областью применения всего приложения.  
«Биржевой бот» ориентирован на рынок ценных бумаг. Целевая аудитория - люди, напрямую связанные с инвестициями. Проект потенциально может развиться в «боевого» бота, позволяющего людям автоматизировать торговлю на рынках ценных бумаг, криптовалют и т. п. логируемых позиций.

# ЦЕЛЬ ИСПЫТАНИЙ

Целью испытаний является проверка корректности выполнения программой функций, перечисленных в разделе «Требования к программе».

# ТРЕБОВАНИЯ К ПРОГРАММЕ

Программа должна соответствовать следующим требованиям, указанным в документе «API-модуль программы “Биржевой бот”. Техническое задание».

# Требования к функциональным характеристикам

# Требования к составу выполняемых функций программы

**а) API- модуль для получения информации о ценах акций**В первую очередь необходимо наладить взаимодействие модуля с выбранным API.  
Выбор пал на официальный клиент Tinkoff Invest [8]. Мотивация выбора описана в приложении 2.  
  
Ключевые требования к модулю (влияют на работу других модулей):1. Обработка и предоставление полученной информации в установленных форматах (подробнее о форматах в развернутом списке требований, предоставленном ниже и в п. «Организация входных и выходных данных»)  
2. Эффективность при работе с большим количеством наблюдаемых акций (агрегация ответа сервера не более секунды)  
3. Передача информации о наличии неполадок на стороне сервиса, предоставляющего информацию о ценах (передача сообщений основному модулю о наличии неполадок)  
4. Проверка входных данных об акциях, поступающих от основного модуля, на корректность и существование ( т.е. решение ситуаций, когда поступает запрос на несуществующие позиции\ запрос на продажу\покупку невозможного количества акций - считаем невозможными те данные, которые не кратны лоту или <0).

5. Восстановление после сбоя (см. «Требования к надежности»)

**Развернутый список требований:**

1. Использование токена для работы с Tinkoff API [9]. Токен генерируется в личном кабинете сайта Тинькофф Инвестиции [8], срок жизни токена – 3 месяца с даты последнего использования. В проекте используется readonly token[7] — он предназначен только для получения различной информации, такой как состояние портфеля, расписание торгов различных торговых площадок, текущие котировки, исторические данные и т.п., а выставление торговых поручений невозможно.
2. Предоставление модулям программы полного перечня акций.  
   Предоставляется: время запроса – общее для всех акций, figi\имя\цена\валюта\статус покупки\статус продажи\лот – для каждой акции.
3. Предоставление перечня акций, доступных к покупке.  
   Предоставляется: время запроса – общее для всех акций, figi\имя\цена\валюта\статус покупки\статус продажи\лот – для каждой акции. Отличие от предыдущего пункта - статус покупки у акций в предоставляемом перечне всегда true.
4. Предоставление перечня акций, доступных к продаже.  
   Предоставляется: время запроса – общее для всех акций, figi\имя\цена\валюта\статус покупки\статус продажи\лот – для каждой акции. Отличие от предыдущего пункта - статус продажи у акций в предоставляемом перечне всегда true.
5. Предоставление цены для конкретной акции за произвольный период в прошлом.  
   Произвольный период и шаг подаются как входные данные, шаг – минута\5минут\15 минут\час\день.  
   Предоставляется список время-цена за запрашиваемый период.
6. Получение информации об акции по ее названию.  
   Требование, существующее для выполнения запросов информации по конкретной акции – либо получение информации (цена\валюта\статус покупки\продажи\лот) на момент запроса, т. е. актуальной, либо получение информации по конкретной акции за произвольный период в прошлом.  
     
   **б) API-модуль, эмулирующий работу торгового API**

В этом модуле логируется запись транзакций.

1. Совершение виртуальных операций купли/продажи на бирже в соответствии с принятым основным модулем решением (в рамках проекта совершение виртуальной операции мы осуществляем через логирование транзакций)
2. Предоставление информации о совершенных операциях в формате [время- позиция - количество единиц - тип операции(купля/продажа) - цена за единицу]
3. Запись истории транзакций в файл [статус транзакции: success\failure, информация об операции – см.п.2 текущего списка].
4. Обработка некорректных\неактуальных данных – передача оповещения основному модулю, продолжение работы программы (за счет игнорирования данных, признанных некорректными).  
   Статус транзакции логируется как failure.

# Организация хранения данных

Модуль-эмулятор, описанный в п. 3.1.1(б), сохраняет логи записей транзакций в отдельный json-файл. Файл именуется датой, когда проводились транзакции. Новый день – новый файл.

# Организация входных данных API- модуля для получения информации о ценах акций

Запросами можно считать весь развернутый список требований к модулю, описанный в п.3.1.1(а).  
Для их выполнения необходимо передавать токен Tinkoff API [7], который уже указан в программе.  
Кроме того, в списке присутствует два типа запросов, для которых формат запроса необходимо уточнить:

1. Запрос текущей информации для наблюдаемых акций в реальном времени, формат запроса [название акции]
2. Запрос цен для конкретной акции за произвольный период в прошлом, формат запроса [название акции - начало периода – конец периода -шаг: шаг устанавливается в рамках дозволенного tinkoff.invest.CandleInterval – не указан\минута\5 минут\15 минут\час\день]

# Организация входных данных модуля, эмулирующего работу торгового API

Программа принимает на вход следующий запрос:  
Решение, касающееся совершения виртуальной операции купли\продажи - сообщение, генерируемое модулем принятия решений в формате [акцию N купить/продать в такое-то время T].

# Организация выходных данных API- модуля для получения информации о ценах акций

Программа предоставляет модулям следующие данные:

1. Передача полного перечня акций (получаемого с помощью API) в формате [время запроса, (figi, имя, цена, валюта, статус покупки, статус продажи, лот)] - словарь.
2. Передача перечня акций, доступных к покупке, в формате  
   [время запроса, (figi, имя, цена, валюта, статус покупки, статус продажи, лот)] – словарь.
3. Передача перечня акций, доступных к продаже, в формате  
   [время запроса, (figi, имя, цена, валюта, статус покупки, статус продажи, лот)] – словарь.
4. Передача информации о конкретной акции за запрошенный период в формате

[время-цена] – лист, размеры определяются запрашиваемым периодом.

1. Передача сообщения о наличии неполадок на стороне сервиса, предоставляющего информацию об акциях. Если возникает одна из установленных ошибок, возвращается сообщение 'Internal error'.
2. Передача сообщения о невозможности обработать запрос (в связи с его отсутствием запрашиваемых позиций\некорректным форматом запроса) - передается сообщение 'No such share name'.

# Организация выходных данных модуля, эмулирующего работу торгового API

Выходными данными модуля считаются json-файлы, содержащие логи транзакций.  
Логируются как успешные, так и неуспешные транзакции. При неуспешных транзакциях пишется причина неудачи.  
*Пример:  
Обработка случая с запросом на покупку\продажу невозможного\* количества позиций.  
Статус транзакции логируется как failure, выводится соответствующее сообщение:*

*"status": "failure",*

*"share": operation['share'],*

*"operation": operation['type'],*

*"message": incorrect quantity to {operation['type']}, lot is {share\_info['lot']}*

*\* невозможным количеством считаем число, которое меньше нуля\не кратно лоту.*

# Требования к временным характеристикам

Скорость работы системы зависит от скорости интернет-подключения и от объема поступающих данных. Эффективной работа системы будет считаться, если агрегация ответа сервера длится не более секунды.

# Требования к интерфейсу

Требования к интерфейсу не предъявляются.

# Требования к надежности

### **3.4.1. Требования к обеспечению надежного (устойчивого) функционирования программы**

При любых действиях пользователя и при любых входных данных, система не должна завершаться из-за ошибки.

### **3.4.2. Требования к корректности данных**

Программа должна обеспечивать проверку корректности входных данных.

Некорректными входными данными считаются те данные, которые не соответствуют описанному формату запроса. Форматы запроса описаны в пунктах «Организация входных данных».

### **3.4.3. Отказы из-за некорректных действий оператора**

Отказ может возникнуть только при условии повреждения исходного кода.

# ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

# Состав программной документации

1. «API-модуль программы “Биржевой бот”». Техническое задание (ГОСТ 19.201-78[2]);
2. «API-модуль программы “Биржевой бот”». Программа и методика испытаний (ГОСТ 19.301-78[3]);
3. «API-модуль программы “Биржевой бот”». Текст программы (ГОСТ 19.401-78[4]);
4. «API-модуль программы “Биржевой бот”». Пояснительная записка (ГОСТ 19.404-79[5]);
5. «API-модуль программы “Биржевой бот”». Руководство оператора (ГОСТ 19.505-79[6]);

# Специальные требования к программной документации

1. Документы к программе должны быть выполнены в соответствии с ГОСТ 19.106-78[1] и ГОСТами к каждому виду документа (см. п. 5.1.);
2. Пояснительная записка должна быть загружена в систему Антиплагиат через Smart LMS «НИУ ВШЭ». Лист, подтверждающий загрузку пояснительной записки, сдается в учебный офис вместе со всеми материалами не позже, чем за день до защиты курсовой работы;
3. Техническое задание и пояснительная записка, титульные листы других документов должны быть подписаны руководителем разработки и исполнителем Документация и программа сдается в электронном виде в формате .pdf или .docx. в архиве формата .zip или .rar;
4. Все документы перед защитой курсовой работы должны быть загружены в информационно-образовательную среду НИУ ВШЭ систему Smart LMS (Smart Learning Management System), в личном кабинете дисциплина - «Курсовой проект, 2 курс ПИ», одним архивом.

# СРЕДСТВА И ПОРЯДОК ИСПЫТАНИЙ

# Технические средства, используемые во время испытаний

Работа программы была протестирована на ОС Windows 10 со следующими характеристиками:

1) Процессор: Intel Core i7-1065G7 CPU 1.30GHz 1.50 GHz

2) Объем ОЗУ 16 GB

3) 64-разрядная архитектура

4) NVIDIA GeForce MX 350

Минимальным требованием для функционирования программы считается наличие ПК и доступа к интернету.

# Программные средства, используемые во время испытаний

Для работы с API-модулем программы “Биржевой бот” разработчику необходимы:

* 1. Предустановленный Python версии 3.9 или выше (основной язык разработки)
  2. Клиент для работы с Tinkoff API - tinkoff.invest[9]
  3. Библиотека для работы с акторами - thespian.actors[11]
  4. Доступ к интернету.

Для функционирования системы требуются следующие программные совместимости:

1. Предустановленный Python версии 3.9 или выше
2. Доступ к жесткому диску
3. Доступ к интернету
4. Unix-система (рекомендована в контексте использования программы в проекте «Биржевой бот», поскольку остальные модули программы тестируются именно на ней)

# Порядок проведения испытаний

Испытания должны проводиться в следующем порядке:

1. Проверка требований к программной документации;
2. Проверка требований к функциональным характеристикам;
3. Проверка требований к временным характеристикам;
4. Проверка требований к надежности.

# 6. МЕТОДЫ ИСПЫТАНИЙ

Испытания представляют собой процесс установки соответствия программы и программной документации заданным требованиям.

Программа «API-модуль программы “Биржевой бот”» поставляется в виде архива. Помимо необходимых для функционирования приложения файлов, в том числе исполняемых, архив содержит все необходимые документы.

# 6.1 Испытание требований к программной документации

Состав программной документации проверяется визуально, проверяется наличие полного комплекта документов в системе Smart LMS. Также, документация проверяется визуально на соответствие нормам ГОСТ. Все документы удовлетворяют представленным требованиям.

6.2 Испытание требований к функциональным характеристикам  
1.Испытания модуля-эмулятора, отвечающего за логирование транзакций, произведены успешно – записываются как успешные, так и не удачные транзакции, файл с датой создается верно.  
Ниже будут предоставлены различные испытания API- модуля и положительная реакция эмулятора (запись принятия решений, запись ошибок на стороне пользователя\программиста и сервиса, запись статуса транзакций, запись транзакций за один день в один файл, генерация файла)  
подтверждает его успешное прохождение испытаний требований к выходным данным.  
 **2**. Испытание работы токена Tinkoff API:

Когда указан существующий токен (сгенерированный в личном кабинете Tinkoff Invest[8]), сервис предоставляет все запрашиваемые данные.  
Когда токен указан неверно, обрабатывается и логируется ошибка на стороне сервиса:

**Изображение выглядит как текст

Автоматически созданное описание***Рисунок 1 – Ошибка авторизации*2. При условии отсутствия ошибок на стороне сервиса предоставляется полный список акций, точно так же можно получить информацию о конкретной акции:  


*Рисунок 2 – Предоставление информации о запрашиваемой акции*Надо заметить, что на рисунке предоставлен не лог, поскольку программа логирует именно транзакции(или попытки их совершить).

Ниже изображен лог ситуации, когда без подключения к интернету попытались запросить информацию об акции и произвести транзакцию с ее участием:

*Изображение выглядит как текст

Автоматически созданное описание*

*Рисунок 3 – Ошибка подключения. Причина: отсутствие подключения к интернету*

3.Испытание запроса на совершение операции – покупка конкретной акции, количество 1, цена 0.01.   
Модуль-эмулятор сделал запись о совершенной транзакции – она успешна, поскольку указана существующая акция и допустимое количество.

*Изображение выглядит как текст

Автоматически созданное описание  
Рисунок 4 – Запись об успешной покупке акции*

5.Если ошибка случилась на стороне пользователя\программиста – запрошена несуществующая акция - запишется соответствующий лог:

*Изображение выглядит как текст

Автоматически созданное описание*

*Рисунок 5 – Ошибка ввода: некорректное название акции*

Основываясь на данных наблюдениях, можно сделать вывод, что требования к функциональным характеристикам полностью удовлетворены (однако возможна последующая балансировка).

# 6.3 Испытание требований к временным характеристикам

Проведены измерения выполнения запроса полного перечня акций – агрегация ответа сервера длилась ~ 0.845с, что в целом удовлетворяет установленным рамкам эффективной работы системы.

# 6.4. Испытание требований к надежности

По результатам испытаний требований к функциональным характеристикам видно, что программа обеспечивает проверку корректности входных данных и не завершается аварийным способом.

Таким образом, требования к надежности выполнены.

**ПРИЛОЖЕНИЕ 1**

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
2. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
3. ГОСТ 19.301-77 Программа и методика испытаний. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
4. ГОСТ 19.401-78 Текст программы. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
5. ГОСТ 19.404-78 Пояснительная записка. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
6. ГОСТ 19.505-78 Руководство оператора. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
7. Tinkoff Invest Api Token// [Электронный ресурс] - Режим доступа: свободный, URL: <https://tinkoff.github.io/investAPI/token/> (5 мая 2022)
8. Тинькофф инвестиции // [Электронный ресурс] - Режим доступа: свободный, URL: <https://www.tinkoff.ru/invest/> (5 мая 2022)
9. Тинькофф клиент API // [Электронный ресурс] - Режим доступа: свободный, URL: <https://tinkoff.github.io/invest-python/> (5 мая 2022)
10. Thespian Actors User`s Guide// [Электронный ресурс] - Режим доступа: свободный, URL: <https://thespianpy.com/doc/using.html> (5 мая 2022)
11. Thespian Actors Library// [Электронный ресурс] - Режим доступа: свободный, URL: <https://github.com/kquick/Thespian>(5 мая 2022)
12. Finage.co.uk - предоставляет информацию в реальном времени для всех бирж в Соединенных Штатах и для внебиржевых рынков //[Электронный ресурс] - Режим доступа: свободный, URL: <http://finage.co.uk/> (5 мая 2022)
13. **Yahoo Finance API** // [Электронный ресурс] - Режим доступа: свободный, URL: [https://www.yahoofinanceapi.com](https://www.yahoofinanceapi.com/) (5 мая 2022)
14. Alpha Vantage API // [Электронный ресурс] - Режим доступа: свободный, URL:  
    <https://www.alphavantage.co/documentation/> (5 мая 2022)
15. EXANTE API // [Электронный ресурс] - Режим доступа: свободный, URL:  
    <https://exante.eu/ru/technology/> (5 мая 2022)

**ПРИЛОЖЕНИЕ 2**

# СРАВНЕНИЕ СУЩЕСТВУЮЩИХ API И ОБОСНОВАНИЕ ВЫБОРА

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | [Finage.co.uk](http://finage.co.uk/)[12] | [Alpha Vantage API](https://www.alphavantage.co/documentation/) [14] | [EXANTE](https://exante.eu/ru/technology/)[15] | [Yahoo Finance API](https://www.yahoofinanceapi.com)[13] | Tinkoff API[9] |
| Бесплатная подписка | + | + | + | + | + |
| Верификация данных пользователя (удостоверения личности) для использования сервиса | - | - | + | - | + |
| Лимиты на запрос (обращения к API) | + | \_ | \_ | + | - |
| Подробная документация и примеры использования | + | + | \_ | \_ | + |
| Возможность получения текущих цен акций | + | \_ | + | + | + |
| Возможность получения данных за прошедший период | + | + | + | \_ | + |
| Возможность использования для реальной торговли на бирже (личное пользование) | - | - | + | - | + |

Цветом в табличке обозначена польза для проекта. Зеленый – подходит, красный – не подходит. Насыщенный красный – категорически не подходит. По цветовой схеме видно, что под все предъявляемые требования идеально подошел Tinkoff API.   
  
Следует отметить, что этот выбор не лишен недостатков: для его использования необходим открытый в банке инвестиционный счет, а публичный сервис на основе TINKOFF INVEST API сделать невозможно, т. к. это противоречит установленным условиям использования (TINKOFF INVEST API предоставляется только клиентам Тинькофф без права на ретрансляцию данных).

# ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | | |
| Номера листов (страниц) | | | | | Всего листов (страниц в докум.) | № документа | Входящий № сопроводительного докум. и дата | Подп. | Дата |
| Изм. | Измененных | Замененных | Новых | Аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |